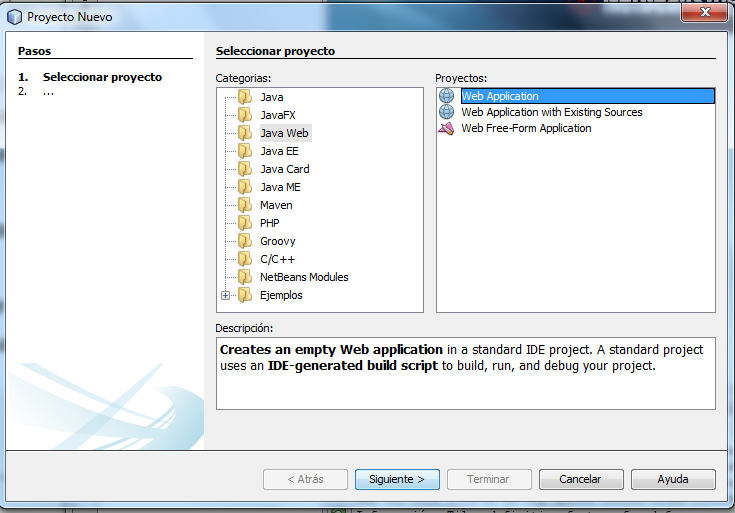
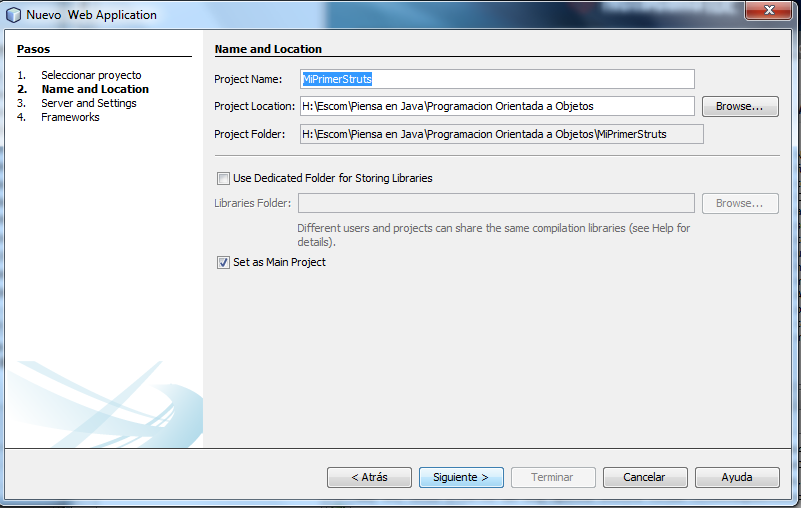
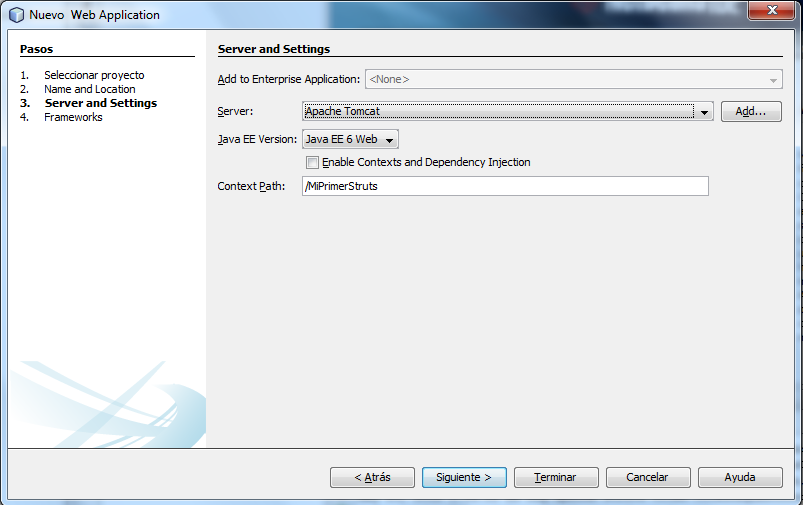
Introducción al Framework de Struts Web

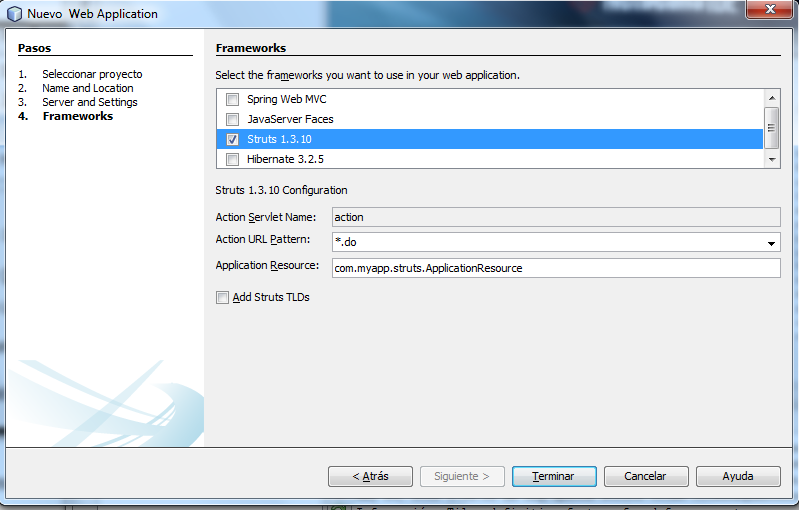
Crear un nuevo proyecto en la interfaz NetBeans, desde la categoría "Web" y el tipo "Web aplication".

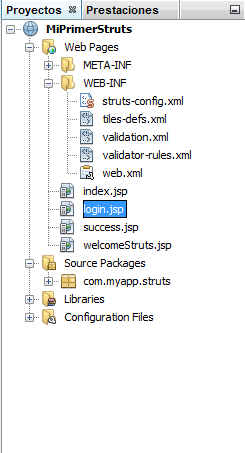
Seleccionar un nombre cualquiera para nuestro nuevo proyecto

Seleccionaremos un tipo de servidor para la realización del mismo, elegir Apache Tomcat



En la última pestaña para la creación del proyecto, elegiremos la opción de Struts 1.3.10



Se crean 2 archivos con extension.jsp, el primero de nombre login.jsp, y el otro de nombre success.jsp.

Para crear estos dos archivos, es necesario dar clic derecho al nodo de nuestro proyecto, desde la ventana "Projects" de

la interfaz de Netbeans, y seleccionar New->jsp.

Dentro del archivo login.jsp va el siguiente código

<%@ taglib uri="http://struts.apache.org/tags-bean"

prefix="bean" %>

<%@ taglib uri="http://struts.apache.org/tags-html"

prefix="html" %>

<%@page contentType="text/html" pageEncoding="UTF-

8"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01

Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type"

content="text/html; charset=UTF-8">

<link rel="stylesheet" type="text/css"

href="stylesheet.css">

<title>Login Form</title>

</head>

<body>

<html:form action="/login">

<table border="0">

<tbody>

<tr>

<td colspan="2">

<bean:write

name="LoginForm" property="error" filter="false"/>

&nbsp;</td>

</tr>

<tr>

<td>Enter your name:</td>

<td><html:text

property="name" /></td>

</tr>

<tr>

<td>Enter your email:</td>

<td><html:text

property="email" /></td>

</tr>

<tr>

<td></td>

<td><html:submit

value="Login" /></td>

</tr>

</tbody>

</table>

</html:form>

</body>

</html>

Dentro del archivo success.jsp va el siguiente código

<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<link rel="stylesheet" type="text/css" href="stylesheet.css">

<title>Login Success</title>

</head>

<body>

<h1>Congratulations!</h1>

<p>You have successfully logged in.</p>

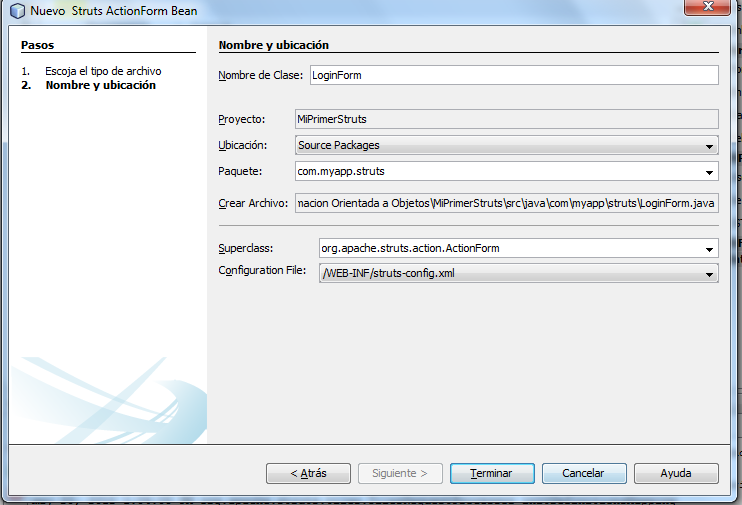
<p>Your name is: <bean:write name="LoginForm" property="name" />.</p>

<p>Your email address is: <bean:write name="LoginForm" property="email" />.</p>

</body>

</html>

Ahora crearemos el ActionForm. Se da clic secundario sobre el nodo de tu proyecto, y da clic en New--Other y selecciona dentro de la categoría de struts el tipo de archivo Struts ActionForm Bean. En el nombre de la clase, teclea LoginForm y selecciona com.myapp.struts en package.



Dentro del archivo LoginForm.java va el siguiente código

package com.myapp.struts;

import javax.servlet.http.HttpServletRequest;

import org.apache.struts.action.ActionErrors;

import org.apache.struts.action.ActionMapping;

import org.apache.struts.action.ActionMessage;

/\*\*

\*

\* @author Edwin

\*/

public class LoginForm extends

org.apache.struts.action.ActionForm {

private String name;

private String email;

// error message

private String error;

public String getError() {

return error;

}

public void setError() {

this.error =

"<span style='color:red'>Please

provide valid entries for both fields</span>";

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

private int number;

/\*\*

\* @return

\*/

public String getName() {

return name;

}

/\*\*

\* @param string

\*/

public void setName(String string) {

name = string;

}

/\*\*

\* @return

\*/

public int getNumber() {

return number;

}

/\*\*

\* @param i

Núñez Flores Gerardo de Jesús

\*/

public void setNumber(int i) {

number = i;

}

/\*\*

\*

\*/

public LoginForm() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* This is the action called from the Struts

framework.

\* @param mapping The ActionMapping used to

select this instance.

\* @param request The HTTP Request we are

processing.

\* @return

\*/

public ActionErrors validate(ActionMapping

mapping, HttpServletRequest request) {

ActionErrors errors = new ActionErrors();

if (getName() == null || getName().length()

< 1) {

errors.add("name", new

ActionMessage("error.name.required"));

// TODO: add 'error.name.required' key

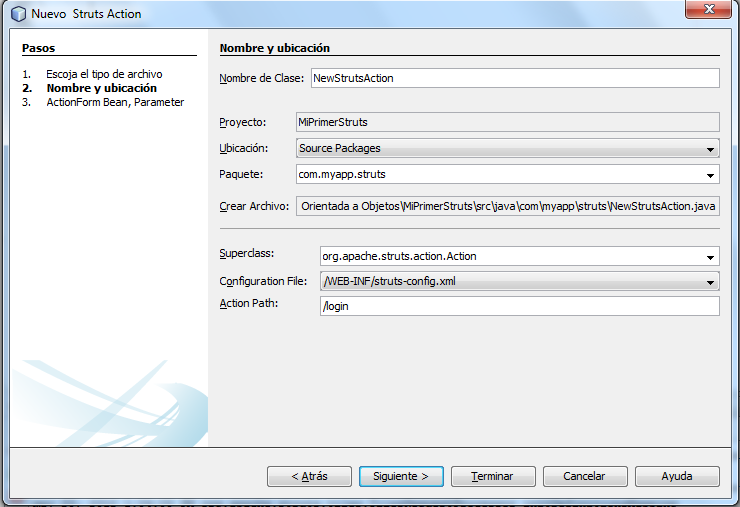
to your resources

}

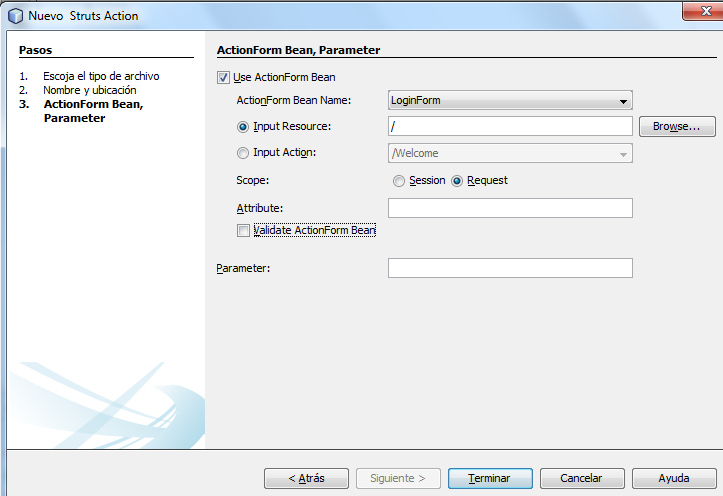
return errors;

}

Ahora crearemos un archivo Struts Action, para optimizar nuestras interacciones entre archivos. Para eso, daremos clic en nuevo desde el menu del boton secundario del nodo de nuestro proyecto. Seleccionaremos la categoria Structs., y seleccionaremos el tipo de archivo Structs Action. En el nombre de la clase pondremos LoginAction, seleccionamos com.myapp.struts como paquete, y en el action path, pondremos /login.



Eliminaremos el contenido del campo "InputResource", definiremos a "Scope" como Request, y deseleccionaremos el campo de "Validate ActionForm Bean" en la siguiente ventana.



Dentro del archivo LoginAction.java va el siguiente código

package com.myapp.struts;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

public class LoginAction extends org.apache.struts.action.Action {

private static final String SUCCESS = "success";

private final static String FAILURE = "failure";

@Override

public ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response)

throws Exception {

LoginForm formBean = (LoginForm) form;

String name = formBean.getName();

String email = formBean.getEmail();

if ((name == null) || // name parameter does not exist

email == null || // email parameter does not exist

name.equals("") || // name parameter is empty

email.indexOf("@") == -1) { // email lacks '@'

formBean.setError();

return mapping.findForward(FAILURE);

}

return mapping.findForward(SUCCESS);

}

}

Dentro del archivo struts-config.xml va el siguiente código

<?xml version="1.0" encoding="UTF-8" ?>

<!DOCTYPE struts-config PUBLIC

"-//Apache Software Foundation//DTD Struts Configuration 1.3//EN"

"http://jakarta.apache.org/struts/dtds/struts-config\_1\_3.dtd">

<struts-config>

<form-beans>

<form-bean name="LoginForm" type="com.myapp.struts.LoginForm"/>

</form-beans>

<global-exceptions>

</global-exceptions>

<global-forwards>

<forward name="welcome" path="/Welcome.do"/>

</global-forwards>

<action-mappings>

<action name="LoginForm" path="/login" scope="request" type="com.myapp.struts.LoginAction"

validate="false">

<forward name="success" path="/WEB-INF/success.jsp"/>

<forward name="failure" path="/login.jsp"/>

</action>

<action path="/Welcome" forward="/welcomeStruts.jsp"/>

</action-mappings>

<controller processorClass="org.apache.struts.tiles.TilesRequestProcessor"/>

<message-resources parameter="com/myapp/struts/ApplicationResource"/>

<plug-in className="org.apache.struts.tiles.TilesPlugin" >

<set-property property="definitions-config" value="/WEB-INF/tiles-defs.xml" />

<set-property property="moduleAware" value="true" />

</plug-in>

<plug-in className="org.apache.struts.validator.ValidatorPlugIn">

<set-property

property="pathnames"

value="/WEB-INF/validator-rules.xml,/WEB-INF/validation.xml"/>

</plug-in>

</struts-config>

Ahora para definir al archivo login.jsp como inicio para el proyecto. Para hacer esto, Dentro del archivo web.xml va el siguiente código

<?xml version="1.0" encoding="UTF-8"?>

<web-app version="3.0" xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd">

<context-param>

<param-name>javax.faces.PROJECT\_STAGE</param-name>

<param-value>Development</param-value>

</context-param>

<servlet>

<servlet-name>Faces Servlet</servlet-name>

<servlet-class>javax.faces.webapp.FacesServlet</servlet-class>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>Faces Servlet</servlet-name>

<url-pattern>/faces/\*</url-pattern>

</servlet-mapping>

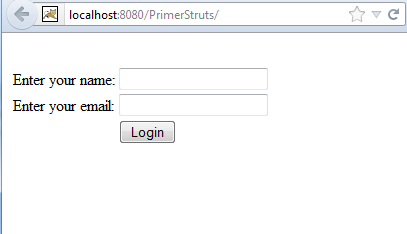
<welcome-file-list>

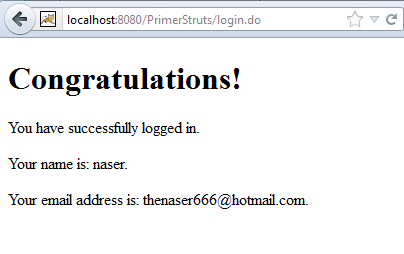
<welcome-file>faces/index.xhtml</welcome-file>

</welcome-file-list>

</web-app>

Y el resultado de nuestra aplicación es el siguiente.





Y así podemos ver también como nos pide datos correctos para poder entrar.

